[0032] In another variant, the signature information 110 also contains information about the signatory's public key 118. This information 118 could a reference to where a third party can obtain the public key. Alternatively, the information 118 could be the signatory's public key or a digital certificate containing the signatory's public key 118. If the signatory utilized more than one public key to generate a digital signature, then each public key could be included along with information identifying which digital signatures were generated using which of the public keys.

[0033] In an embodiment, the universal signature object 100 also includes use-permission information 130. The use-permission information 130 indicates how a version or versions 102-104 of the digital data 200 can be utilized. For example, the use-permission information can indicate that a particular user may only have certain rights, such as read-only or view-only rights. Alternatively, the use-permission can give various users varied levels of access to a version 102-104 of the digital data 200. The universal-signature-object viewer 600, which will be explained in more detail below, utilizes this use-permission information.

[0034] In an embodiment, the universal signature object 100 also includes a universal-signature-object viewer (USO viewer) 600, which is an executable file that can utilize the universal signature object 100 to generate information from or related to the universal signature object 100. The universal-signature-object viewer will be described in more detail below.

[0035] In an embodiment, the universal signature object 100 also includes a signing program 400, which is an executable file used to generate a universal signature object 100 or to append a digital signature to an existing universal signature object 100. The signing program 400 will be described in more detail below.

[0036] Figure 2 depicts an embodiment of a system capable of generating and utilizing a universal signature object 100. Figure 2 depicts a signing program 400 connected via a network connection 308 to a timing source 210, a transaction server 220, and a verification service 230. The network could be a local area network or a wide area network. In one embodiment, the signing program 400 connects to the timing source 210, the transaction server 220, and the verification service 230 via the Internet 240. In alternate embodiments, the timing source 210, the transaction server 220, and/or the verification service 230 reside on the same computer as the signing program 400 or within the same local area network. It shall also be noted that the timing source 210, the transaction server 220, and the verification service 230 can be different functions performed by a single entity.

[0037] Figure 2 depicts a private key 202 and a corresponding public key 204 of a signatory accessible by the signing program 400. Also shown in Figure 2, the digital data 200 is used by the signing program 400 in generating a universal signature object 100, which a USO viewer 600 utilizes to provide a user with information related to or derived from the universal signature object 100. The signing program 400 can be executed on a computer system, such as a personal computer or workstation.

[0038] Figure 3 illustrates a computer system 300 wherein a processor 302 executes software instructions and interacts with other system components. A storage device 304 coupled to the processor 302 provided long-term storage of data and software programs and may be implemented as a hard disk drive or other suitable mass storage devices. A network interface 306 coupled to the processor 302 connects 308 the computer system to a network. A display device 310 coupled to the processor 302 displays text and graphics under the control of the processor 302. An input device 312, such as a mouse and or keyboard, is coupled to the processor 302 and facilitates user control of the system 300. An addressable memory 312 coupled to the processor 302 stores software instructions 320, 322 to be executed by the processor 302 and is implemented using a combination of standard memory devices such as random access memory ("RAM") and read only memory ("ROM") devices. In one embodiment, the memory 312 stores a number of software objects or modules, for example, a first application 320 and a second application 322. The applications 320, 322, individually or collectively, could represent the signing program 400 and the USO viewer 600.

[0039] Throughout this discussion, modules or means are described as separate functional units. This is done for clarity of explanation. In different implementations, various means or modules may be combined and integrated into a single software application or device. Alternatively, various means or modules may be distributed into several software applications or devices. The modules or means can also be implemented in software, hardware, firmware, or any combination thereof.

[0040] Figure 4 represents an embodiment of the signing program 400, which could be an application 320, 322 operating on system 300. Signing program 400 comprises a key-accessing means 402, a key-verification means 404, transaction tracking means 406, a universal-signature-object generating means 408, and a timestamping means 410. These means or modules in the signing program 400 interface with the processor 302 as represented by arrow 316. Key-accessing means 402 accesses the private 202 and public 204 keys of a signatory. Key-verification means 404 verifies the authenticity of the private and public key pair 202, 204 (respectively). The USO generating means 408 generates a universal signature object 100 or appends a digital signature to an existing universal signature object 100. The generation of the universal signature object 100 will be described in more detail with respect to Figure 5. The transaction tracking means 406 interacts with a transaction server in order to provide an audit trail or to archive a digital signature or a USO 100.

[0041] Figure 5 depicts an embodiment of a method for generating a universal signature object 100 as part of the system depicted in Figure 2. The key-accessing means 402 of the signing program 400 accesses 502 the private 202 and public 204 keys of a signatory 500. The signatory 500 can supply the private-public key pair 202, 204 to the signing program 400 in a number of ways. In one embodiment, the private and public key pair 202, 204 is stored on the storage device 304 and accessed by the signing program 400 through the processor 302. Alternatively, the key pair is stored on a network and accessed through the network interface